



(12) 发明专利

(10) 授权公告号 CN 114500552 B

(45) 授权公告日 2022.10.18

(21) 申请号 202210087877.3

H04L 67/62 (2022.01)

(22) 申请日 2022.01.25

H04L 67/63 (2022.01)

(65) 同一申请的已公布的文献号

审查员 丁炜

申请公布号 CN 114500552 A

(43) 申请公布日 2022.05.13

(73) 专利权人 北京秒如科技有限公司

地址 100089 北京市海淀区中关村东路18

号1号楼10层A-1110-001号

(72) 发明人 丰国升

(74) 专利代理机构 北京众合诚成知识产权代理

有限公司 11246

专利代理师 陈国强

(51) Int. Cl.

H04L 67/1004 (2022.01)

H04L 67/14 (2022.01)

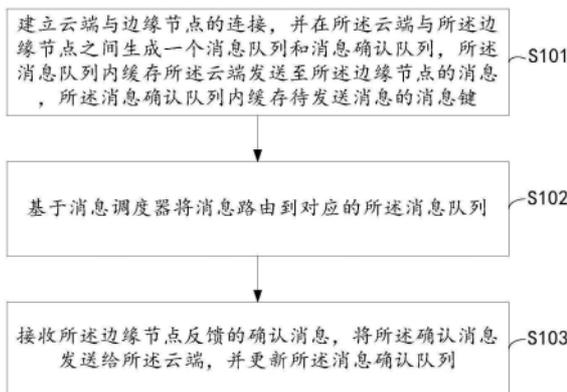
权利要求书3页 说明书8页 附图1页

(54) 发明名称

边缘计算场景下的云边消息可靠性传输方法及装置

(57) 摘要

本发明提供一种边缘计算场景下的云边消息可靠性传输方法及装置,涉及边缘计算技术,包括:建立云端与边缘节点之间的连接,并在所述云端与所述边缘节点之间生成一个消息队列和消息确认队列,所述消息队列内缓存所述云端发送至所述边缘节点的消息,所述消息确认队列内缓存待发送消息的消息键;基于消息调度器将消息路由到对应的所述消息队列;接收所述边缘节点反馈的确认消息,将所述确认消息发送给所述云端,并更新所述消息确认队列。本发明可以更有效且更少资源占用的将信息从云端发送到边缘端。



1. 一种边缘计算场景下的云边消息可靠性传输方法,其特征在于,包括:

建立云端与边缘节点的连接,并在所述云端与所述边缘节点之间生成一个消息队列和消息确认队列,所述消息队列内缓存所述云端发送至所述边缘节点的消息,所述消息确认队列内缓存待发送消息的消息键;

基于消息调度器将消息路由到对应的所述消息队列;

接收所述边缘节点反馈的确认消息,将所述确认消息发送给所述云端,并更新所述消息确认队列;

还包括:

若所述云端在预设时间内未收到所述确认消息,基于消息重试策略重发对应的消息;

在基于消息重试策略重发对应的消息失败之后,生成失败信息发送给告警模块;

所述消息重试策略包括:

固定间隔策略,所述固定间隔策略重试的时间间隔为固定值;

指数退让策略,所述指数退让策略重试的时间间隔呈指数增加;

增量间隔策略,所述增量间隔策略重试的时间间隔增量递增;

随机化策略,所述随机化策略重试的时间间隔为随机值;

还包括:

获取消息不可靠类型,基于预设策略对所述不可靠类型的消息进行处理;

其中,所述不可靠类型包括云端重启、边缘节点重启、边缘节点删除、边缘节点离线、边缘节点长时间离线、云端发送过程中丢失、边缘节点接收过程中丢失、云端消息积压;

所述预设策略包括:

若所述不可靠类型为云端重启,获取所有消息,依次遍历消息是否过期,若过期从消息队列中删除,确定消息队列中均为有效消息;

若所述不可靠类型为边缘节点重启,将所述边缘节点发送的心跳消息转发给所述云端,若所述消息队列被删除,重新生成所述消息队列,基于重新生成的所述消息队列发送消息;若所述不可靠类型为边缘节点删除,移除对应的所述消息队列;

若所述不可靠类型为边缘节点离线,暂停消息发送,缓存消息等待所述边缘节点上线;

若所述不可靠类型为边缘节点长时间离线,释放所述消息队列;

若所述不可靠类型为云端发送过程中丢失,所述云端触发所述消息重试策略进行重试;

若所述不可靠类型为边缘节点接收过程中丢失,所述云端触发所述消息重试策略进行重试,在边缘节点收到消息后,比对存储DB内对应消息的version是否一致,若是,返回确认消息;

若所述不可靠类型为云端消息积压,执行预热策略;

所述预热策略包括:

获取边缘节点的启动时间 t_1 ;

使用当前时间减去边缘节点的启动时间 t_1 ,计算出边缘节点运行的时间 t_2 ;

根据 t_2 的时间动态计算权重确定最终每次发送到边缘节点消息的数量,直到达到默认设置的预热时间阈值,云端正常发送消息。

2. 根据权利要求1所述的方法,其特征在于,所述消息队列内缓存所述云端发送至所述

边缘节点的消息,包括:

基于所述消息队列缓存所述云端发送至所述边缘节点的消息键,并响应发送指令获取所述消息键对应的消息。

3. 根据权利要求1所述的方法,其特征在于,更新所述消息确认队列,包括:将所述确认消息对应的消息键从所述消息确认队列中删除。

4. 根据权利要求1所述的方法,其特征在于,在基于消息调度器将消息路由到对应的所述消息队列之前,还包括:

判断所述消息队列内的消息是否为最新;

若是,丢弃当前待调度消息,若否将当前待调度消息路由到对应的所述消息队列。

5. 一种边缘计算场景下的云边消息可靠性传输装置,其特征在于,包括:

连接模块,用于建立云端与边缘节点的连接,并在所述云端与所述边缘节点之间生成一个消息队列和消息确认队列,所述消息队列内缓存所述云端发送至所述边缘节点的消息,所述消息确认队列内缓存待发送消息的消息键;

调度模块,用于基于消息调度器将消息路由到对应的所述消息队列;

更新模块,用于接收所述边缘节点反馈的确认消息,将所述确认消息发送给所述云端,并更新所述消息确认队列;

还包括:

若所述云端在预设时间内未收到所述确认消息,基于消息重试策略重发对应的消息;

在基于消息重试策略重发对应的消息失败之后,生成失败信息发送给告警模块;

所述消息重试策略包括:

固定间隔策略,所述固定间隔策略重试的时间间隔为固定值;

指数退让策略,所述指数退让策略重试的时间间隔呈指数增加;

增量间隔策略,所述增量间隔策略重试的时间间隔增量递增;

随机化策略,所述随机化策略重试的时间间隔为随机值;

还包括:

获取消息不可靠类型,基于预设策略对所述不可靠类型的消息进行处理;

其中,所述不可靠类型包括云端重启、边缘节点重启、边缘节点删除、边缘节点离线、边缘节点长时间离线、云端发送过程中丢失、边缘节点接收过程中丢失、云端消息积压;

所述预设策略包括:

若所述不可靠类型为云端重启,获取所有消息,依次遍历消息是否过期,若过期从消息队列中删除,确定消息队列中均为有效消息;

若所述不可靠类型为边缘节点重启,将所述边缘节点发送的心跳消息转发给所述云端,若所述消息队列被删除,重新生成所述消息队列,基于重新生成的所述消息队列发送消息;若所述不可靠类型为边缘节点删除,移除对应的所述消息队列;

若所述不可靠类型为边缘节点离线,暂停消息发送,缓存消息等待所述边缘节点上线;

若所述不可靠类型为边缘节点长时间离线,释放所述消息队列;

若所述不可靠类型为云端发送过程中丢失,所述云端触发所述消息重试策略进行重试;

若所述不可靠类型为边缘节点接收过程中丢失,所述云端触发所述消息重试策略进行

重试,在边缘节点收到消息后,比对存储DB内对应消息的version是否一致,若是,返回确认消息;

若所述不可靠类型为云端消息积压,执行预热策略;

所述预热策略包括:

获取边缘节点的启动时间 t_1 ;

使用当前时间减去边缘节点的启动时间 t_1 ,计算出边缘节点运行的时间 t_2 ;

根据 t_2 的时间动态计算权重确定最终每次发送到边缘节点消息的数量,直到达到默认设置的预热时间阈值,云端正常发送消息。

边缘计算场景下的云边消息可靠性传输方法及装置

技术领域

[0001] 本发明涉及边缘计算技术,尤其涉及一种边缘计算场景下的云边消息可靠性传输方法及装置。

背景技术

[0002] 在边缘信息的传输机制中,存在三种类型的消息传输机制,至少一次 (At-Least-Once),至多一次 (At-Most-Once),有且仅有一次 (Exactly-Once);使用“Exactly-Once”非常昂贵并且性能最差,尽管它提供了没有消息丢失或重复的保证传递。

[0003] 现有技术中,现有云端发送到边缘节点的消息部分为“最多一次”的传输方案,消息都存储于队列中,然后通过socket将数据发送到边缘节点。由于云边网络环境或者边缘节点的离线等因素很容易导致云端发送到边缘节点的消息丢失,从而造成云边数据不一致的状态;还有采用至少一次的传输方案,但是没有考虑到边缘场景的复杂性和系统的资源占用。

[0004] 因此,如何更有效且更少资源占用的将信息从云端发送到边缘端,成为了急需解决的问题。

发明内容

[0005] 本发明实施例提供一种边缘计算场景下的云边消息可靠性传输方法及装置,可以更有效且更少资源占用的将信息从云端发送到边缘端。

[0006] 本发明实施例的第一方面,提供一种边缘计算场景下的云边消息可靠性传输方法,包括:

[0007] 建立云端与边缘节点的连接,并在所述云端与所述边缘节点之间生成一个消息队列和消息确认队列,所述消息队列内缓存所述云端发送至所述边缘节点的消息,所述消息确认队列内缓存待发送消息的消息键;

[0008] 基于消息调度器将消息路由到对应的所述消息队列;

[0009] 接收所述边缘节点反馈的确认消息,将所述确认消息发送给所述云端,并更新所述消息确认队列。

[0010] 可选地,在第一方面的一种可能实现方式中,所述消息队列内缓存所述云端发送至所述边缘节点的消息,包括:

[0011] 基于所述消息队列缓存所述云端发送至所述边缘节点的消息键,并响应发送指令获取所述消息键对应的消息。

[0012] 可选地,在第一方面的一种可能实现方式中,更新所述消息确认队列,包括:

[0013] 将所述确认消息对应的消息键从所述消息确认队列中删除。

[0014] 可选地,在第一方面的一种可能实现方式中,在基于消息调度器将消息路由到对应的所述消息队列之前,还包括:

[0015] 判断所述消息队列内的消息是否为最新;

- [0016] 若是,丢弃当前待调度消息,若否将当前待调度消息路由到对应的所述消息队列。
- [0017] 可选地,在第一方面的一种可能实现方式中,还包括:
- [0018] 若所述云端在预设时间内未收到所述确认消息,基于消息重试策略重发对应的消息;
- [0019] 在基于消息重试策略重发对应的消息失败之后,生成失败信息发送给告警模块。
- [0020] 可选地,在第一方面的一种可能实现方式中,所述消息重试策略包括:
- [0021] 固定间隔策略,所述固定间隔策略重试的时间间隔为固定值;
- [0022] 指数退让策略,所述指数退让策略重试的时间间隔呈指数增加;
- [0023] 增量间隔策略,所述增量间隔策略重试的时间间隔增量递增;
- [0024] 随机化策略,所述随机化策略重试的时间间隔为随机值。
- [0025] 可选地,在第一方面的一种可能实现方式中,还包括:
- [0026] 获取消息不可靠类型,基于预设策略对所述不可靠类型的消息进行处理;
- [0027] 其中,所述不可靠类型包括云端重启、边缘节点重启、边缘节点删除、边缘节点离线、边缘节点长时间离线、云端发送过程中丢失、边缘节点接收过程中丢失、云端消息积压。
- [0028] 可选地,在第一方面的一种可能实现方式中,所述预设策略包括:
- [0029] 若所述不可靠类型为云端重启,获取所有消息,依次遍历消息是否过期,若过期从消息队列中删除,确定消息队列中均为有效消息;
- [0030] 若所述不可靠类型为边缘节点重启,将所述边缘节点发送的心跳消息转发给所述云端,若所述消息队列被删除,重新生成所述消息队列,基于重新生成的所述消息队列发送消息;
- [0031] 若所述不可靠类型为边缘节点删除,移除对应的所述消息队列;
- [0032] 若所述不可靠类型为边缘节点离线,暂停消息发送,缓存消息等待所述边缘节点上线;
- [0033] 若所述不可靠类型为边缘节点长时间离线,释放所述消息队列;
- [0034] 若所述不可靠类型为云端发送过程中丢失,所述云端触发所述消息重试策略进行重试;
- [0035] 若所述不可靠类型为边缘节点接收过程中丢失,所述云端触发所述消息重试策略进行重试,在边缘节点收到消息后,比对存储DB内对应消息的version是否一致,若是,返回确认消息;
- [0036] 若所述不可靠类型为云端消息积压,执行预热策略。
- [0037] 可选地,在第一方面的一种可能实现方式中,所述预热策略包括:
- [0038] 获取边缘节点的启动时间 t_1 ;
- [0039] 使用当前时间减去边缘节点的启动时间 t_1 ,计算出边缘节点运行的时间 t_2 ;
- [0040] 根据 t_2 的时间动态计算权重确定最终每次发送到边缘节点消息的数量,直到达到默认设置的预热时间阈值,云端正常发送消息。
- [0041] 本发明实施例的第二方面,提供一种边缘计算场景下的云边消息可靠性传输装置,包括:
- [0042] 连接模块,用于建立云端与边缘节点连接,并在所述云端与所述边缘节点之间生成一个消息队列和消息确认队列,所述消息队列内缓存所述云端发送至所述边缘节点的

消息,所述消息确认队列内缓存待发送消息的消息键;

[0043] 调度模块,用于基于消息调度器将消息路由到对应的所述消息队列;

[0044] 更新模块,用于接收所述边缘节点反馈的确认消息,将所述确认消息发送给所述云端,并更新所述消息确认队列。

[0045] 本发明实施例的第三方面,提供一种边缘计算场景下的云边消息可靠性传输设备,包括:存储器、处理器以及计算机程序,所述计算机程序存储在所述存储器中,所述处理器运行所述计算机程序执行本发明第一方面及第一方面各种可能涉及的所述方法。

[0046] 本发明实施例的第四方面,提供一种可读存储介质,所述可读存储介质中存储有计算机程序,所述计算机程序被处理器执行时用于实现本发明第一方面及第一方面各种可能涉及的所述方法。

[0047] 技术效果:

[0048] 1、本发明针对边缘计算场景,提出了更可靠的云边消息传输方法和装置,通过设计云端存储、边缘侧存储、传输过程多次重试等机制保证了云边消息的可靠性,通过设计独立的节点消息队列,实现了对边缘节点消息的创建、更新、删除的管理,保证了最新的消息发送到边缘节点。

[0049] 2、本发明针对边缘节点的特性,设计了不同的消息重发策略,极大的增强了消息发送到边缘节点的可能性,减少了整个系统资源的占用;

[0050] 3、与现有方案相比,本发明中的节点消息队列和消息确认队列都只保存了消息键,减少了内存占用;

[0051] 4、本发明中,分析了云边场景下可能导致不可靠传输的场景,并针对性的提出了解决方案。

附图说明

[0052] 图1是本发明实施例提供的一种边缘计算场景下的云边消息可靠性传输方法的流程示意图;

[0053] 图2是本发明实施例提供的一种边缘计算场景下的云边消息可靠性传输装置的结构示意图;

[0054] 图3是本发明实施例提供的一种边缘计算场景下的云边消息可靠性传输设备的硬件结构示意图。

具体实施方式

[0055] 为使本发明实施例的目的、技术方案和优点更加清楚,下面将结合本发明实施例中的附图,对本发明实施例中的技术方案进行清楚、完整地描述,显然,所描述的实施例仅仅是本发明一部分实施例,而不是全部的实施例。基于本发明中的实施例,本领域普通技术人员在没有做出创造性劳动前提下所获得的所有其他实施例,都属于本发明保护的范围。

[0056] 本发明的说明书和权利要求书及上述附图中的术语“第一”、“第二”、“第三”“第四”等(如果存在)是用于区别类似的对象,而不必用于描述特定的顺序或先后次序。应该理解这样使用的数据在适当情况下可以互换,以便这里描述的本发明的实施例能够以除了在这里图示或描述的那些以外的顺序实施。

[0057] 应当理解,在本发明的各种实施例中,各过程的序号的大小并不意味着执行顺序的先后,各过程的执行顺序应以其功能和内在逻辑确定,而不应对本发明实施例的实施过程构成任何限定。

[0058] 应当理解,在本发明中,“包括”和“具有”以及他们的任何变形,意图在于覆盖不排他的包含,例如,包含了一系列步骤或单元的过程、方法、系统、产品或设备不必限于清楚地列出的那些步骤或单元,而是可包括没有清楚地列出的或对于这些过程、方法、产品或设备固有的其它步骤或单元。

[0059] 应当理解,在本发明中,“多个”是指两个或两个以上。“和/或”仅仅是一种描述关联对象的关联关系,表示可以存在三种关系,例如,A和/或B,可以表示:单独存在A,同时存在A和B,单独存在B这三种情况。字符“/”一般表示前后关联对象是一种“或”的关系。“包含A、B和C”、“包含A、B、C”是指A、B、C三者都包含,“包含A、B或C”是指包含A、B、C三者之一,“包含A、B和/或C”是指包含A、B、C三者中任1个或任2个或3个。

[0060] 应当理解,在本发明中,“与A对应的B”、“与A相对应的B”、“A与B相对应”或者“B与A相对应”,表示B与A相关联,根据A可以确定B。根据A确定B并不意味着仅仅根据A确定B,还可以根据A和/或其他信息确定B。A与B的匹配,是A与B的相似度大于或等于预设的阈值。

[0061] 取决于语境,如在此所使用的“若”可以被解释成为“在……时”或“当……时”或“响应于确定”或“响应于检测”。

[0062] 下面以具体地实施例对本发明的技术方案进行详细说明。下面这几个具体的实施例可以相互结合,对于相同或相似的概念或过程可能在某些实施例不再赘述。

[0063] 参见图1,是本发明实施例提供的一种边缘计算场景下的云边消息可靠性传输方法的流程示意图,图1所示方法的执行主体可以是软件和/或硬件装置。本申请的执行主体可以包括但不限于以下中的至少一个:用户设备、网络设备等。其中,用户设备可以包括但不限于计算机、智能手机、个人数字助理(Personal Digital Assistant,简称:PDA)及上述提及的电子设备等。网络设备可以包括但不限于单个网络服务器、多个网络服务器组成的服务器组或基于云计算的由大量计算机或网络服务器构成的云,其中,云计算是分布式计算的一种,由一群松散耦合的计算机组成的一个超级虚拟计算机。本实施例对此不做限制。包括步骤S101至步骤S104,具体如下:

[0064] S101,建立云端与边缘节点的连接,并在所述云端与所述边缘节点之间生成一个消息队列和消息确认队列,所述消息队列内缓存所述云端发送至所述边缘节点的消息,所述消息确认队列内缓存待发送消息的消息键。

[0065] 具体的,在进行消息传输之前,必须先建立起云端与边缘节点之间的连接。在实际应用中,边缘节点可以设置有多个,且多个边缘节点均可以连接到云端,与云端之间进行数据传输。例如,可以设置有10个边缘节点来与云端进行连接,进行数据传输。

[0066] 在云端与边缘节点连接之后,本发明会在云端和边缘节点之间生成一个消息队列和消息确认队列,来进行消息的传输。

[0067] 示例性的,针对每一个边缘节点,云端对应一个消息队列和消息确认队列,例如10个边缘节点,云端会有10个消息队列和消息确认队列一一对应。

[0068] 其中,消息队列内缓存有云端发送至边缘节点的消息。

[0069] 在一些实施例中,为了减少资源占用,本方案的消息队列内只存储云端发送至边

缘节点的消息键,并响应发送指令获取消息键对应的消息,即在准备好发送时,才获取到对应的消息数据。

[0070] 其中,消息确认队列内缓存待发送消息的消息键。

[0071] 需要说明的是,消息确认队列主要用来存储发送到边缘节点的消息键,当收到边缘节点返回的确认消息时,从消息确认队列中删除消息键,并将消息的最新状态更新到存储DB中。

[0072] 本方案通过消息确认队列来确定消息被发送,防止出现消息遗漏或者丢失发送的情况出现。

[0073] S102,基于消息调度器将消息路由到对应的所述消息队列。

[0074] 可以理解的是,消息调度器即用于调度待发送的消息,本方案中,消息调度器用于将消息路由到对应的所述消息队列。例如,消息1需要通过消息队列2发送,那么消息调度器会将消息1调度到对应的消息队列2中,以通过消息队列2将消息发送出去。

[0075] 需要说明的是,一个节点对应一个队列,是想要将消息发送到指定节点,节点的关系和队列是一一对应的,消息是路由到节点对应的队列。

[0076] 在一些实施例中,本方案为了确保消息队列内的消息为最新,且防止消息重复,采用以下方案:

[0077] 在基于消息调度器将消息路由到对应的所述消息队列之前,还包括:

[0078] 判断所述消息队列内的消息是否为最新;若是,丢弃当前待调度消息,若否将当前待调度消息路由到对应的所述消息队列。

[0079] 可以理解的是,本方案在利用消息调度器将消息路由到对应的所述消息队列之前,需要先比较消息队列内的消息是否是最新的,如果消息队列中的消息最新,则丢弃掉当前的消息,否则将该消息更新至消息队列内。

[0080] S103,接收所述边缘节点反馈的确认消息,将所述确认消息发送给所述云端,并更新所述消息确认队列。

[0081] 具体的,在消息实现调度完成后,处于待发送状态,将消息从消息队列中取出,发送到对应的边缘节点即可完成信息的传输。

[0082] 在边缘节点接收到云端发送的消息后,为了确认边缘节点接收到了消息,边缘节点会生成一个确认消息发送给云端,告知云端已接收到消息,可以确保消息传输成功,提高消息的可靠性传输。

[0083] 此外,本方案还会对消息确认队列进行更新,以告知消息确认队列传输完成,若传输未完成,不会对消息确认队列进行更新。

[0084] 其中,更新所述消息确认队列,可以是将所述确认消息对应的消息键从所述消息确认队列中删除。

[0085] 上述实施例提供的边缘计算场景下的云边消息可靠性传输方法,可以更有效且更少资源占用的将信息从云端发送到边缘端。

[0086] 在上述实施例的基础上,如果云端在指定的时间内没有收到来自边缘节点反馈的确认消息,本方案采用以下方法将重发该消息n次,以让消息可靠传输。

[0087] 若所述云端在预设时间内未收到所述确认消息,基于消息重试策略重发对应的消息;在基于消息重试策略重发对应的消息失败之后,生成失败信息发送给告警模块。

[0088] 可以理解的是,本方案采用消息重试策略来重发对应的消息,并在重发多次失败之后,将失败信息发送给告警模块进行告警。

[0089] 其中,消息重试策略包括固定间隔策略、指数退让策略、增量间隔策略以及随机化策略。

[0090] 其中,所述固定间隔策略重试的时间间隔为固定值,例如:每3秒进行重试操作;所述指数退让策略重试的时间间隔呈指数增加;例如:3秒、12秒、30秒后进行重试操作;所述增量间隔策略重试的时间间隔增量递增;例如:3秒、7秒、13秒后进行重试操作;所述随机化策略重试的时间间隔为随机值。

[0091] 需要说明的是,上述的重试策略都可以包含随机化,配合其他策略使用。

[0092] 在现有的消息传输中,还会出现各种消息不可靠的场景,例如,不可靠类型包括云端重启、边缘节点重启、边缘节点删除、边缘节点离线、边缘节点长时间离线、云端发送过程中丢失、边缘节点接收过程中丢失、云端消息积压,出现以上场景时,会导致消息发送不可靠,本方案为了解决此问题,采用以下方案。

[0093] 针对不可靠类型为云端重启:

[0094] 本方案获取所有消息,依次遍历消息是否过期,若过期从消息队列中删除,确定消息队列中均为有效消息。可以理解的是,云端重启时,将检查是否有过期的消息,将过期的消息删除,保证队列中都是有效消息。

[0095] 针对不可靠类型为边缘节点重启:

[0096] 本方案将所述边缘节点发送的心跳消息转发给所述云端,若所述消息队列被删除,重新生成所述消息队列,基于重新生成的所述消息队列发送消息。可以理解的是,当边缘节点重启时,边缘节点发送心跳信息到云端,云端感知到边缘节点已经上线后,如果缓存的消息队列因为节点长久离线而删除,则重新创建一个新的消息队列,云端将有效消息缓存到此队列,然后发送到边缘节点。

[0097] 针对不可靠类型为边缘节点删除:

[0098] 本方案移除对应的所述消息队列。可以理解的是,当一个边缘节点从云端删除时,云端会移除对应的消息队列。具体可以是监听删除事件,更新消息的最新状态,然后移除消息队列。

[0099] 针对不可靠类型为边缘节点离线:

[0100] 本方案暂停消息发送,缓存消息等待所述边缘节点上线;可以理解的是,边缘节点在离线状态时,云端检测到断连之后,将停止无意义的发送和重试,缓存消息等待边缘节点恢复可用。

[0101] 针对不可靠类型为边缘节点长时间离线:

[0102] 本方案释放所述消息队列。可以理解的是,边缘节点长时间离线时,释放消息队列,减少内存占用。其中,初始边缘节点的长时间离线时间从配置中读取,后续的判别方法为通过此边缘节点历史离线情况结合同组同类型的边缘节点离线情况,以及云端当前的负载情况,通过专家打分法确定最终的离线时间阈值。云端判断边缘节点离线时间如果大于计算的最长离线时间,则释放消息队列。

[0103] 针对不可靠类型为云端发送过程中丢失:

[0104] 本方案云端触发所述消息重试策略进行重试。可以理解的是,发送过程中丢失时,

云端可以触发重试机制,按照上述实施例中的重试机制进行重试。

[0105] 针对不可靠类型为边缘节点接收过程中丢失:

[0106] 本方案所述云端触发所述消息重试策略进行重试,在边缘节点收到消息后,比对存储DB内对应消息的version是否一致,若是,返回确认消息;可以理解的是,接收过程中丢失时,此时边缘节点侧已经处理了消息,云端触发重试机制,按照上述机制进行重试,边缘节点接收到重复消息时,从存储DB中获取的对应消息的version,version一致,则不作处理,返回确认消息。

[0107] 针对不可靠类型为云端消息积压:

[0108] 本方案执行预热策略。可以理解的是,由于边缘节点长时间离线等原因,可能导致云端的消息队列积压。大量积压的消息批量发送到边缘节点时,由于边缘节点重启后服务运行状态还没有达到最佳,边缘节点很可能被压垮宕机,所以需要预热。预热的策略是:

[0109] 获取边缘节点的启动时间 t_1 ;

[0110] 使用当前时间减去边缘节点的启动时间 t_1 ,计算出边缘节点运行的时间 t_2 ;

[0111] 根据 t_2 的时间动态计算权重确定最终每次发送到边缘节点消息的数量,直到达到默认设置的预热时间阈值,云端正常发送消息。

[0112] 本方案采用“至少一次”的传输策略,使用反馈的确认消息来保证消息可靠到达边缘节点,使用version保证多次消息重发边缘端处理的幂等性问题,边缘节点使用存储DB保存最新从云端接收的消息,并且在不同的场景下设置了不同的机制来最大限度保证消息的可靠性。

[0113] 需要说明的是,本专利创新的提出的边缘计算场景下的云边消息可靠性传输方法,带有一定的普适性,不限于本专利中所举例的边缘计算场景下,某些受限环境云边传输消息也可适用,例如,也适用于边缘节点发送消息到云端的场景。

[0114] 参见图2,是本发明实施例提供的一种边缘计算场景下的云边消息可靠性传输装置的结构示意图,该边缘计算场景下的云边消息可靠性传输装置包括:

[0115] 连接模块,用于建立云端与边缘节点的连接,并在所述云端与所述边缘节点之间生成一个消息队列和消息确认队列,所述消息队列内缓存所述云端发送至所述边缘节点的消息,所述消息确认队列内缓存待发送消息的消息键;

[0116] 调度模块,用于基于消息调度器将消息路由到对应的所述消息队列;

[0117] 更新模块,用于接收所述边缘节点反馈的确认消息,将所述确认消息发送给所述云端,并更新所述消息确认队列。

[0118] 图2所示实施例的装置对应地可用于执行图1所示方法实施例中的步骤,其实现原理和技术效果类似,此处不再赘述。

[0119] 参见图3,是本发明实施例提供的一种边缘计算场景下的云边消息可靠性传输设备的硬件结构示意图,该边缘计算场景下的云边消息可靠性传输设备30包括:处理器31、存储器32和计算机程序;其中

[0120] 存储器32,用于存储所述计算机程序,该存储器还可以是闪存(flash)。所述计算机程序例如是实现上述方法的应用程序、功能模块等。

[0121] 处理器31,用于执行所述存储器存储的计算机程序,以实现上述方法中设备执行的各个步骤。具体可以参见前面方法实施例中的相关描述。

[0122] 可选地,存储器32既可以是独立的,也可以跟处理器31集成在一起。

[0123] 当所述存储器32是独立于处理器31之外的器件时,所述设备还可以包括:

[0124] 总线33,用于连接所述存储器32和处理器31。

[0125] 本发明还提供一种可读存储介质,所述可读存储介质中存储有计算机程序,所述计算机程序被处理器执行时用于实现上述的各种实施方式提供的方法。

[0126] 其中,可读存储介质可以是计算机存储介质,也可以是通信介质。通信介质包括便于从一个地方向另一个地方传送计算机程序的任何介质。计算机存储介质可以是通用或专用计算机能够存取的任何可用介质。例如,可读存储介质耦合至处理器,从而使处理器能够从该可读存储介质读取信息,且可向该可读存储介质写入信息。当然,可读存储介质也可以是处理器的组成部分。处理器和可读存储介质可以位于专用集成电路(Application Specific Integrated Circuits,简称:ASIC)中。另外,该ASIC可以位于用户设备中。当然,处理器和可读存储介质也可以作为分立组件存在于通信设备中。可读存储介质可以是只读存储器(ROM)、随机存取存储器(RAM)、CD-ROM、磁带、软盘和光数据存储设备等。

[0127] 本发明还提供一种程序产品,该程序产品包括执行指令,该执行指令存储在可读存储介质中。设备的至少一个处理器可以从可读存储介质读取该执行指令,至少一个处理器执行该执行指令使得设备实施上述的各种实施方式提供的方法。

[0128] 在上述设备的实施例中,应理解,处理器可以是中央处理单元(英文:Central Processing Unit,简称:CPU),还可以是其他通用处理器、数字信号处理器(英文:Digital Signal Processor,简称:DSP)、专用集成电路(英文:Application Specific Integrated Circuit,简称:ASIC)等。通用处理器可以是微处理器或者该处理器也可以是任何常规的处理器等。结合本发明所公开的方法的步骤可以直接体现为硬件处理器执行完成,或者用处理器中的硬件及软件模块组合执行完成。

[0129] 最后应说明的是:以上各实施例仅用以说明本发明的技术方案,而非对其限制;尽管参照前述各实施例对本发明进行了详细的说明,本领域的普通技术人员应当理解:其依然可以对前述各实施例所记载的技术方案进行修改,或者对其中部分或者全部技术特征进行等同替换;而这些修改或者替换,并不使相应技术方案的本质脱离本发明各实施例技术方案的范围。

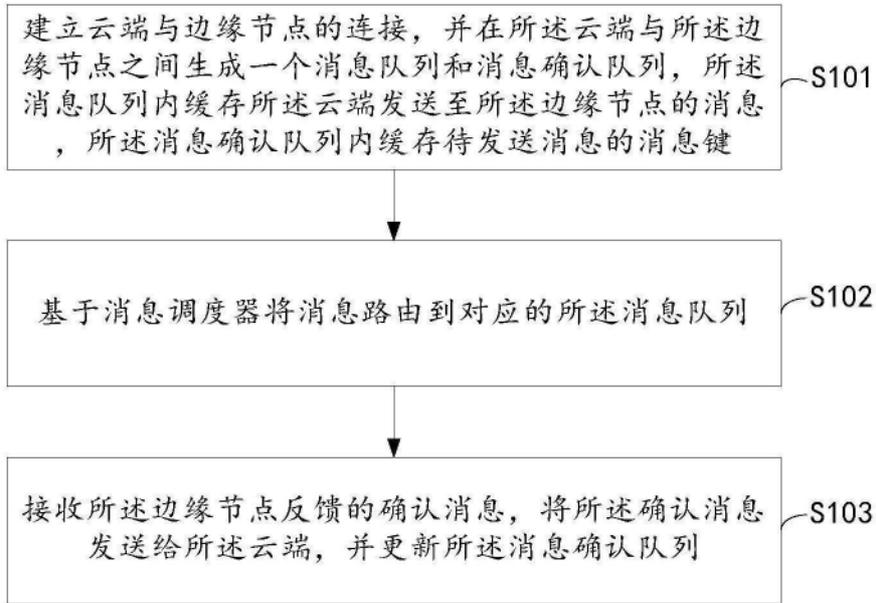


图1

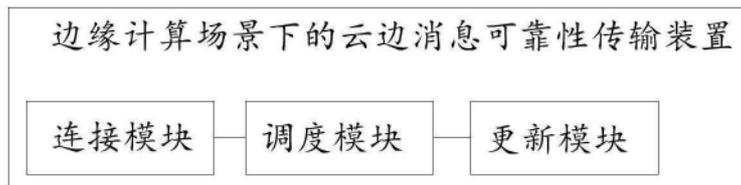


图2

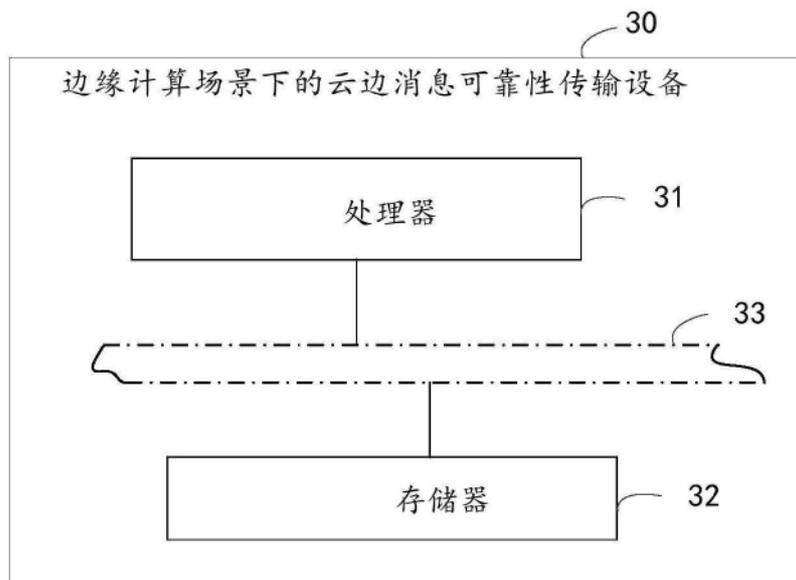


图3